11-08-00

A

Please type a plus sign (+) inside this box [+]

PTO/SB/05 (12/97)
Approved for use through 09/30/00. OMB 0651-0032
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b)

Attorney Docket No. 003022.P019X

Total Pages 5

First Named Inventor or Application Identifier Vance C. Bjorn

Express Mail Label No. EL143569313US

ADDRESS TO: **Assistant Commissioner for Patents**
Box Patent Application
Washington, D. C. 20231

APPLICATION ELEMENTS
See MPEP chapter 600 concerning utility patent application contents.

1. X Fee Transmittal Form
(Submit an original, and a duplicate for fee processing)

2. X Specification (Total Pages 32)
(preferred arrangement set forth below)
- Descriptive Title of the Invention
- Cross References to Related Applications
- Statement Regarding Fed sponsored R & D
- Reference to Microfiche Appendix
- Background of the Invention
- Brief Summary of the Invention
- Brief Description of the Drawings (if filed)
- Detailed Description
- Claims
- Abstract of the Disclosure

3. X Drawings(s) (35 USC 113) (Total Sheets 8)

4. ___ Oath or Declaration (Total Pages ___)

a. ___ Newly Executed (Original or Copy)

b. ___ Copy from a Prior Application (37 CFR 1.63(d))
(for Continuation/Divisional with Box 17 completed) **(Note Box 5 below)**

i. ___ DELETIONS OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).

5. ___ Incorporation By Reference (useable if Box 4b is checked)
The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.

6. ___ Microfiche Computer Program (Appendix)

7. _____ Nucleotide and/or Amino Acid Sequence Submission
(if applicable, all necessary)
a. ____ Computer Readable Copy
b. ____ Paper Copy (identical to computer copy)
c. ____ Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

8. _____ Assignment Papers (cover sheet & documents(s))
9. _____ a. 37 CFR 3.73(b) Statement (where there is an assignee)
_____ b. Power of Attorney
10. _____ English Translation Document (if applicable)
11. _____ a. Information Disclosure Statement (IDS)/PTO-1449
_____ b. Copies of IDS Citations
12. _____ Preliminary Amendment
13. _X_ Return Receipt Postcard (MPEP 503) (Should be specifically itemized)
14. _____ a. Small Entity Statement(s)
b. Statement filed in prior application, Status still proper and desired
15. _____ Certified Copy of Priority Document(s) (if foreign priority is claimed)
16. _X_ Other: a copy of the postcard with Certificate of Express Mailing.

17. **If a CONTINUING APPLICATION,** check appropriate box and supply the requisite information:
___ Continuation ____ Divisional _X_ Continuation-in-part (CIP)
of prior application No: 09/538,926

18. **Correspondence Address**
_____ Customer Number or Bar Code Label _____
(Insert Customer No. or Attach Bar Code Label here)
or
_X_ Correspondence Address Below

NAME Judith A. Szepesi, Reg. No. 39,393
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP 11/6/2000
ADDRESS 12400 Wilshire Boulevard
Seventh Floor
CITY Los Angeles STATE California ZIP CODE 90025-1026
Country U.S.A. TELEPHONE (408) 720-8598 FAX (408) 720-9397

Approved for use through 09/30/00. OMB 0651-0032
Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

PTO/SB/17(08-00)
Approved for use through 10/31/2002. OMB 0651-0032
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

FEE TRANSMITTAL FOR FY 2001

TOTAL AMOUNT OF PAYMENT ($) 908.00

Complete if Known:
Application No. To be assigned
Filing Date Herewith
First Named Inventor Vance C. Bjorn
Group Art Unit To be assigned
Examiner Name To be assigned
Attorney Docket No. 003022.P019X

METHOD OF PAYMENT (check one)

1. [X] **The Commissioner is hereby authorized to charge indicated fees and credit any over payments to:**

Deposit Account Number 02-2666
Deposit Account Name _____

[ ] **Charge Any Additional Fee Required Under 37 CFR 1.16 and 1.17**

---

2. X **Payment Enclosed:**
X **Check**
_____ **Money Order**
_____ **Other**

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
101	710	201	355	Utility application filing fee	710.00
106	320	206	160	Design application filing fee	
107	490	207	245	Plant filing fee	
108	710	208	355	Reissue filing fee	
114	150	214	75	Provisional application filing fee	

SUBTOTAL (1) $ 710.00

2. EXTRA CLAIM FEES

			Extra Claims		Fee from below		Fee Paid
Total Claims	31	– 20** =	11	X	18.00	=	198.00
Independent Claims	3	– 3** =	0	X	80.00	=	0
Multiple Dependent						=	

**Or number previously paid, if greater; For Reissues, see below.

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
103	18	203	9	Claims in excess of 20
102	80	202	40	Independent claims in excess of 3
104	270	204	135	Multiple dependent claim, if not paid
109	80	209	40	**Reissue independent claims over original patent
110	18	210	9	**Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) $ 198.00

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
105	130	205	65	Surcharge - late filing fee or oath	
127	50	227	25	Surcharge - late provisional filing fee or cover sheet	
139	130	139	130	Non-English specification	
147	2,520	147	2,520	For filing a request for reexamination	
112	920*	112	920*	Requesting publication of SIR prior to Examiner action	
113	1,840*	113	1,840*	Requesting publication of SIR after Examiner action	
115	110	215	55	Extension for response within first month	
116	390	216	195	Extension for response within second month	
117	890	217	445	Extension for response within third month	
118	1,390	218	695	Extension for response within fourth month	
128	1,890	228	945	Extension for response within fifth month	
119	310	219	155	Notice of Appeal	
120	310	220	155	Filing a brief in support of an appeal	
121	270	221	135	Request for oral hearing	
138	1,510	138	1,510	Petition to institute a public use proceeding	
140	110	240	55	Petition to revive unavoidably abandoned application	
141	1,240	241	620	Petition to revive unintentionally abandoned application	
142	1,240	242	620	Utility issue fee (or reissue)	
143	440	243	220	Design issue fee	
144	600	244	300	Plant issue fee	
122	130	122	130	Petitions to the Commissioner	
123	50	123	50	Petitions related to provisional applications	
126	240	126	240	Submission of Information Disclosure Stmt	
581	40	581	40	Recording each patent assignment per property (times number of properties)	
146	710	246	355	For filing a submission after final rejection (see 37 CFR 1.129(a))	
149	710	249	355	For each additional invention to be examined (see 37 CFR 1.129(b))	
179	710	279	355	Request for Continued Examination (RCE)	
169	900	169	900	Request for expedited examination of a design application	

Other fee (specify) _____

Other fee (specify) _____

SUBTOTAL (3) $ 0

*Reduced by Basic Filing Fee Paid

SUBMITTED BY:

Typed or Printed Name: Judith A. Szepesi

Signature: _____ Date: 11/6/2002

Reg. Number: 39,393 Telephone Number: 408-720-8300

EXPRESS MAIL CERTIFICATE OF MAILING

"Express Mail" mailing label number: EL143569313US
Date of Deposit: November 6, 2000
I hereby certify that I am causing this paper or fee to be deposited with the United States Postal Service "Express Mail Post Office to Addressee" service on the date indicated above and that this paper or fee has been addressed to the Assistant Commissioner for Patents, Washington, D. C. 20231

Conny Willesen
(Typed or printed name of person mailing paper or fee)

Conny Willesen
(Signature of person mailing paper or fee)

11-6-00
(Date signed)

Serial/Patent No.: To be assigned - Filing/Issue Date: ***
Client: Digital Persona, Inc.
Title: A Method And Apparatus For Using A Third Party Authentication Server
BSTZ File No.: 003022.P019X Atty/Secty Initials: JHS/JAS/cvw
Date Mailed: 11-06-00 Docket Due Date: ***

The following has been received in the U.S. Patent & Trademark Office on the date stamped hereon:

- ☐ Amendment/Response (____ pgs.)
- ☐ Appeal Brief (____ pgs.) (in triplicate)
- ☐ Application - Utility (____ pgs., with cover and abstract)
- ☐ Application - Rule 1.53(b) Continuation (____ pgs.)
- ☐ Application - Rule 1.53(b) Divisional (____ pgs.)
- ■ Application - Rule 1.53(b) CIP (32 pgs.)
- ☐ Application - Rule 1.53(d) CPA Transmittal (____ pgs.)
- ☐ Application - Design (____ pgs.)
- ☐ Application - PCT (____ pgs.)
- ☐ Application - Provisional (____ pgs.)
- ☐ Assignment and Cover Sheet
- ■ Certificate of Mailing
- ☐ Declaration & POA (____ pgs.)
- ☐ Disclosure Docs & Orig & Copy of Inventor's Signed Letter (____ pgs)
- ■ Drawings: 8 # of sheets includes 8 figures
- ■ Express Mail No.: EL143569313US
- ☐ ____ Month(s) Extension of Time
- ☐ Information Disclosure Statement & PTO-1449 (__ pgs.)
- ☐ Issue Fee Transmittal
- ☐ Notice of Appeal
- ☐ Petition for Extension of Time
- ☐ Petition for ____
- ■ Postcard
- ☐ Power of Attorney (____ pgs.)
- ☐ Preliminary Amendment (____ pgs.)
- ☐ Reply Brief (____ pgs.)
- ☐ Response to Notice of Missing Parts
- ☐ Small Entity Declaration for Indep. Inventor/Small Business
- ■ Transmittal Letter, in duplicate
- ■ Fee Transmittal, in duplicate

■ Check No. 38786 Amt: $908.00
☐ Check No. ____ Amt: ____

■ Other: a copy of this postcard with Certificate of Express Mailing.

Patent

UNITED STATES PATENT APPLICATION

FOR

A METHOD AND APPARATUS FOR USING A THIRD PARTY AUTHENTICATION SERVER

INVENTORS:

VANCE C. BJORN

PREPARED BY:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1026

(408) 720-8300

ATTORNEY'S DOCKET NO. 003022.P019X

"Express Mail" mailing label number: EL143569313US
Date of Deposit: November 6, 2000
I hereby certify that I am causing this paper or fee to be deposited with the United States Postal Service "Express Mail Post Office to Addressee" service on the date indicated above and that this paper or fee has been addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231

Conny Willesen
(Typed or printed name of person mailing paper or fee)

Conny Willesen
(Signature of person mailing paper or fee)

11-6-00
(Date signed)

A METHOD AND APPARATUS FOR USING A THIRD PARTY AUTHENTICATION SERVER

FIELD OF THE INVENTION

The present invention relates to client-server technology, and more specifically, to using a third party authentication server.

BACKGROUND

As more and more information is moving into electronic form, encryption is becoming more common. One prior art method of encryption is public key encryption -- an encryption scheme in which each person gets a pair of keys, called the public key and the private key. Each person's public key is published while the private key is kept secret. Messages are encrypted using the intended recipient's public key and can only be decrypted using the recipient's private key. Messages are signed using the sender's public key and can only be decrypted using the sender's public key. The need for sender and receiver to share secret information (keys) via some secure channel is eliminated-- all communications involve only public keys, and no private key needs to be transmitted or shared. Public-key cryptography can be used for authentication (digital signatures) as well as for privacy (encryption). Other encryption schemes, such as symmetric key encryption rely on an exchange of keys.

Figure 1 is a diagram of a prior art network. The client 110 connects to a server 130 through network 140. A certification authority 150 provides a private/public key pair for the user. The certification authority 150 further provides certificate 115 to the client. The certificate 115 is a copy of the user's public key, signed by the certification authority 150, to prove its authenticity.

The certificate 115 and the user's private key 120 are stored on the client system 110. Private keys generally are 64 bit numbers or larger and users do not memorize their keys. Because computer systems are rarely truly secure, the key may be taken from a computer system. In order to prevent this, the key may be stored in a password-protected file. However, passwords may be broken. Additionally, the system is only as secure as the least secure level. For one embodiment, the user types in the password 125, to release the private key 120, so the user can use the private key.

Furthermore, generally the keys are stored on a computer system, and are thus connected to the computer system, rather than an actual user. In the prior art, a user could pass to an impostor his or her password, accidentally or on purpose, and that impostor could then "prove" that he or she was the user.

Furthermore, because each user's private key is stored on his or her computer system, administering the keys is difficult.

In addition, a single mistake, i.e. accidentally granting access to an unauthorized user, permanently breaches the security of the private-public key pair, since the private key is revealed.

SUMMARY OF THE INVENTION

A method and apparatus for a third party authentication server is described. The method includes receiving a record ID for a user, and a one-time key generated by the server and encrypted with a user's public key by the server. The method further includes receiving the user's authentication data from the client, and determining if the user's authentication data matches the record ID. If the authentication data matches the record ID, decrypting the one-time key with the user's private key, and returning the decrypted one-time key to the client.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

Figure 1 is a network diagram of a prior art secured access mechanism.

Figure 2 is one embodiment of a network on which the present invention may be implemented.

Figure 3 is a block diagram of one embodiment of a computer system that may be used in conjunction with the present invention.

Figure 4 is a block diagram of one embodiment of a partner site, a client, and an authentication server.

Figure 5 is a diagram of one embodiment of using the authentication server to access a secure partner site.

Figure 6A is a flowchart of one embodiment enabling a client to use the authentication server.

Figure 6B is a flowchart of one embodiment enabling a client to use the third-party authentication with a particular partner site.

Figure 7 is a flowchart of one embodiment of setting up a partner site to use the authentication server.

DETAILED DESCRIPTION

A method and apparatus for a third party authentication server is described. The authentication server described herein enables web services to provide a third party authentication option to their users. For one embodiment, this authentication relies on biometrics. For one embodiment, users use a fingerprint sensor, install it on their system, and within minutes register their fingerprint to access web sites. Many institutions, including banking, financial, healthcare, corporate, and government Intranets and Extranets can benefit from this secure and convenient user authentication mechanism. The system may further be used to unlock a smart card or other secured system. This system is transparent to the user, maintains user privacy, ensures the utmost security of the process, and makes the service very easy to deploy and administer by web services and their customers.

Figure 2 is a block diagram of one embodiment of a network including authentication system. A client 210 is connected to a server 240 through a network 230. If the client 210 wants to log into a secure site on the server 240, the client is prompted by the server 240 to enter the authentication data. This data is sent to the authentication server 220 by the client 210, along with a record ID associated with the particular secure site to which the user is attempting to connect.

For one embodiment, the authentication data is biometric data. In that case, the client system 210 includes a biometric sensor 245. When the user places his or her fingerprint, or other biometric area, over the sensor 245, data is captured. The biometric authentication information is then sent to the authentication server 220.

The authentication server 220 then uses the record ID to determine whether the authentication data matches the registered user. If the user is successfully authenticated, the requested cryptographic function is provided by the authentication server 220. For one embodiment, this cryptographic function is to decrypt a one-time key, provided by server 240, to verify that the user has been successfully authenticated.

For one embodiment, the network 210 may be the Internet. Alternatively, the network 210 may be a local area network (LAN), wide area network (WAN), or another type of network. For one embodiment, the authentication server 220 may be located within the corporation, or the same LAN, or WAN. Thus, a company may install its own authentication server 220, to simplify internal key management.

For another embodiment, the client 210 and authentication server 220 may be on the same computer system. The client 210 may invoke the authentication server 220 when logging on to a server 230 that requires authentication, or whenever cryptographic authentication is needed.

Figure 3 is one embodiment of computer system that may be used with the present invention. It will be apparent to those of ordinary skill in the art, however that other alternative systems of various system architectures may also be used.

The data processing system illustrated in Figure 3 includes a bus or other internal communication means 345 for communicating information, and a processor 340 coupled to the bus 345 for processing information. The system further comprises a random access memory (RAM) or other volatile storage device 350 (referred to as memory), coupled to bus 345 for storing information

and instructions to be executed by processor 340. Main memory 350 also may be used for storing temporary variables or other intermediate information during execution of instructions by processor 340. The system also comprises a read only memory (ROM) and/or static storage device 320 coupled to bus 340 for storing static information and instructions for processor 340, and a data storage device 325 such as a magnetic disk or optical disk and its corresponding disk drive. Data storage device 325 is coupled to bus 345 for storing information and instructions.

The system may further be coupled to a display device 370, such as a cathode ray tube (CRT) or a liquid crystal display (LCD) coupled to bus 345 through bus 365 for displaying information to a computer user. An alphanumeric input device 375, including alphanumeric and other keys, may also be coupled to bus 345 through bus 365 for communicating information and command selections to processor 340. An additional user input device is cursor control device 380, such as a mouse, a trackball, stylus, or cursor direction keys coupled to bus 345 through bus 365 for communicating direction information and command selections to processor 340, and for controlling cursor movement on display device 370.

Another device, which may optionally be coupled to computer system 330, is a communication device 390 for accessing other nodes of a distributed system via a network. The communication device 390 may include any of a number of commercially available networking peripheral devices such as those used for coupling to an Ethernet, token ring, Internet, or wide area network. Note that any or all of the components of this system illustrated in Figure 3 and

associated hardware may be used in various embodiments of the present invention.

It will be appreciated by those of ordinary skill in the art that any configuration of the system may be used for various purposes according to the particular implementation. The control logic or software implementing the present invention can be stored in main memory 350, mass storage device 325, or other storage medium locally or remotely accessible to processor 340. Other storage media may include floppy disks, memory cards, flash memory, or CD-ROM drives.

It will be apparent to those of ordinary skill in the art that the methods and processes described herein can be implemented as software stored in main memory 350 or read only memory 320 and executed by processor 340. This control logic or software may also be resident on an article of manufacture comprising a computer readable medium having computer readable program code embodied therein and being readable by the mass storage device 325 and for causing the processor 340 to operate in accordance with the methods and teachings herein.

The software of the present invention may also be embodied in a handheld or portable device containing a subset of the computer hardware components described above. For example, the handheld device may be configured to contain only the bus 345, the processor 340, and memory 350 and/or 325. The handheld device may also be configured to include a set of buttons or input signaling components with which a user may select from a set of available options. The handheld device may also be configured to include an output apparatus such as a liquid crystal display (LCD) or display element

matrix for displaying information to a user of the handheld device. Conventional methods may be used to implement such a handheld device. The implementation of the present invention for such a device would be apparent to one of ordinary skill in the art given the disclosure of the present invention as provided herein.

Figure 4 is a block diagram of one embodiment of a partner site, a client, and an authentication server. The partner 230, client 240, and authentication server 220 are coupled through networks. For one embodiment, these connections are secure connections.

The client 240 includes a web browser 450 and a network connection 455. The client 240 uses this web browser 450 to receive certain prompts from the partner 230 and the authentication server 220, as will be described below. The client 240 further includes a sensor logic 465, which interfaces with a sensor (not shown) coupled to the client 240. The sensor logic 465 receives an image, such as a biometric image, from the user. For one embodiment, another logic to receive a smart card, to alternative authentication mechanism may also be attached to the client's system.

The client 240 further includes feature extraction logic 470, to extract the features from the biometric data received by sensor logic 470. For another embodiment, the feature extraction logic 470 may be located on the authentication server 220. In that instance, the client 240 sends the actual biometrics, rather than the biometric template extracted from the image.

For one embodiment, the biometric data may be encrypted by the sensor itself. For one embodiment, a challenge response may be used to protect the biometric data.

The replay prevention logic 475 incorporates a nonce, received from the authentication server 220 into the biometric image or the biometric template. The nonce is a one-time number, such as a random number or a number that incorporates data such as the time/date, user IP address, etc. that uniquely identifies the current session. This prevents the reuse of the image/template captured in this session, to establish another secure session.

The encryption logic 460 encrypts communications with the partner 230 and authentication server 220. The encryption logic 460 may also be used to establish secure sessions between the partner 230 and client 240, and between the client 240 and authentication server 220.

The partner 230 includes web pages and scripts 410, which may be displayed to the client 240, using the client's web browser 450. The partner 230 further includes a user interface 415 that is used to interface with the partner 230, to program the partner 230, or to present images/scripts/data to the client 240.

The partner 230 further includes challenge logic 420 to create a challenge, and validation logic 440 to determine whether the third party authentication was successful.

The challenge logic 420, in response to receiving an authentication request, indicating that the user is registered with the authentication server 220, looks up the user's client ID, using the record ID lookup 435. The challenge logic 420 also determines whether additional authentication data is needed from the user, based on the policy associated with the record ID. If the partner site handles additional authentication, such as a password, the challenge logic 420 requests the password, and validates it, prior to passing the record ID to the challenge generator 425.

The challenge generator 425 generates a challenge to be decrypted by the third party authenticator. For one embodiment, the challenge generator 425 generates a long random number, which is encrypted by encryption logic 430, using the public key of the particular user who is about to be authenticated. For one embodiment, the record ID and the challenge are encrypted together. For one embodiment, the policy of the partner is also encrypted with the challenge. This policy may require additional authentication, administered by the authentication server. For example, the policy may require additional biometrics, or a password, administered by the authentication server.

For one embodiment, encryption logic 430 further encrypts the data with the partner key of authentication server 220, to verify that the authentication server 220 used is the "real" authentication server 220. For one embodiment, the partner key is a symmetric key that is passed to the partner 230 when the partner initially registers with the authentication server 220. This registration process is described below. For another embodiment, the partner key may be the public key of the authentication server 220.

The encrypted challenge is then sent out by the partner 230, and a response is awaited. When the response is received, the partner, using the comparison logic 445 determines whether the decrypted challenge received is the actual challenge generated by the challenge generator 425. For one embodiment, the decrypted challenge is actually encrypted with the partner key. This partner key may be the same symmetric key as was used by the partner 230, a different symmetric key, or the public key of the partner 230. If the comparison logic 445 determines that the challenge has been successfully decrypted, e.g. the private

key of the user has been used, the authentication is accepted, and the client 240 is permitted access to the partner 230.

The authentication server 220, which enables this validation process, includes nonce generation logic 480, which generates the nonce used by the client 240 to return the biometric data to the authentication server 220. The nonce, for one embodiment, is a random number.

The authentication server 220 further includes a biometric data comparison logic 485, which compares the biometric data received from the client 240 with the biometric data associated with the particular user. For one embodiment, the user is identified based on the record ID. For one embodiment, the biometric data comparison logic 485 compares two templates. For another embodiment, the biometric data comparison logic 485 further includes a feature extraction logic 470, which generates a template from an image. For yet another embodiment, the template stored in the authentication server 220 may be directly compared with the image received from the client 240.

The policy validation logic 487 determines whether the validation policy of the partner 230 has been fulfilled by the user. As noted above, the policy was included with the challenge, sent by the partner 230 through the client 240. If the validation policy has been fulfilled, i.e. the client 240 has supplied all of the necessary data, the policy validation logic 487 decrypts the challenge, using the decryption logic 490, and returns the decrypted challenge to the client 240. For one embodiment, the decrypted challenge is encrypted with the partner key, prior to being returned.

The decryption logic 490 is used to decrypt communication between the client 240 and the authentication server 220. The decryption logic 490 may use

one or more of the partner key(s), the user's private key, as well as the partner's public key to safely communicate with the other parts of this system.

The authentication server 220 may further include partner registration logic 493, to permit partners to register with the system. For one embodiment, enabling the service includes modifying registration/log-on code, to enable the request for third-party authorization, adding fields to the existing user record database, and installing executables that permit the challenge response mechanism. Furthermore, the partner 240 and authentication server 220 may exchange the partner key. In terms of the partner 230 illustrated, the challenge logic 420 and validation logic 440 are added, and the web pages/scripts 410 are updated and/or replaced to interact appropriately with the authentication server 220. For one embodiment, the partner registration logic 493 may be located on separate server(s).

The authentication server 220 may further include client registration logic 496. Client registration logic 496 prompts the user to install the biometric sensor, if that is not yet installed. The client registration logic 496, for one embodiment, further uploads an installation program that permits the client 240 to register their biometric data. The client registration logic 496 further generates an entry in the database for the new user, and generates a public key/private key pair for the user. For one embodiment, the public key is further certified by a certification authority. For one embodiment, the certification authority may be an external certification authority, such as VeriSign. For another embodiment, the certification authority may be an internal certification authority within the authentication server 220.

The client registration logic 496 further includes logic to pass the public key (for one embodiment, certified) and record ID to the client 240, to be passed on to the partner 230. For one embodiment, the client registration logic 496 may be located on a separate server(s).

Note that the partner 230 and the authentication server 220 do not communicate directly. All communication goes through the client 240.

Figure 5 is a diagram of one embodiment of using the authentication server to access a secure partner site. The authentication server 220 includes a database 510 in which information about those clients that are registered with the authentication server 220 are stored. For one embodiment, database 510 includes a client ID, or record ID 515, which identifies the client. For one embodiment, the client ID 515 is randomly generated at the time the client registers with the authentication server 220.

Associated with a client ID 515 is a biometric template 520. The biometric template 520 is captured during registration. For one embodiment, the biometric template 520 may include multiple fingers. For one embodiment, the biometric template 520 is the processed biometric data. For example, for a fingerprint, the biometric template 520 may be a list of minutiae with locations. Alternative template definitions, as is known in the art, may be used. For yet another embodiment, the biometric template 520 may be an actual image of biometric data. In that instance, the authentication server 220 processes the biometric data upon request.

A client private key 525 is further associated with the client ID 515. Upon registration, a public key/private key pair is generated for the client. The public key is distributed, e.g. passed back to the client. However, the private key 525 is

not released by the authentication server 220. For one embodiment, only the authentication server 220 performs actions using this private key. For one embodiment, a copy of the public key is also kept.

The authentication server 220 further includes a temporary database 530. An entry in the temporary database 530 is generated whenever a new session is established with a client. The temporary database entry is maintained only for a limited period of time. For one embodiment, whenever a client session is closed, the temporary database entry 530 is destroyed.

The temporary database 530 includes a client session ID 535, which is generated when a client session is started. The temporary database 530 further includes a client nonce 540. As was described above with respect to Figure 4, the nonce is a security mechanism that prevents replay attacks. The nonce 540 is a temporary mechanism that is used only for a single access.

The client 240 includes a browser 550 that is able to respond to objects. For one embodiment, the browser 550 supports JavaScript or ActiveX controls. This permits the web page to drive actions on the client's system.

The partner site 230 is the site to which the client 240 is attempting to connect. For one embodiment, the partner site 230 is a local smart card. The local smart card is accessed using this authentication mechanism. The smart card has two portions, the portion that provides the challenge, and the locked portion, which is only accessible if the authentication server properly authenticates the user.

The partner site 230 includes a client database 560. The client database 560 includes a number of entries that are present whether or not the authentication server 220 is used. The "existing entries" 562 are supplemented with "third party

authentication related entries" 580. The existing entries 562 include a client's user name 565, a client password 570, and other client specific data 575. Depending on the site, this data 575 may include the client's account numbers, account contents, etc.

The third party authentication related entries 580 include the client ID 585. The client ID 585 is the same client ID 515 that was generated by the authentication server 220, and passed through the client 240. The client public key 590, matching the private key 525 stored in the authentication server 220, is also stored.

A client policy 592 may be defined by the partner site 230. The client policy 592 determines what items are necessary for authentication. For example, the policy may specify that a single biometric identifier from the client is sufficient for authentication. For another embodiment, the policy may include one or more of the following: biometric identifier(s), smart card(s), password(s), etc. For example, for an extremely high security level, the policy may require three separate biometric identifiers (e.g. two fingerprints and a retina scan), as well as a smart card, and a password. The partner site 230 determines the level of security associated with client access.

A client one-time password 594 is generated when a client first requests access, and is used for the third party authentication, as will be described below.

The process of logging on to a secure web site, using third party authentication, is described as follows. The process starts when a client 240 requests a login page from the website of the partner server 230 (message 1). This is driven by the client's browser 550.

In response to this request, the website sends login page to present the logon options. (message 2). This process is driven by the web server of the partner server 230. The log-on options may include the user name, password, and/or third-party security login. For one embodiment, the log-on options are displayed using an HTML/JavaScript logon page. For one embodiment, an active script determines whether the client 240 has the third-party authentication control installed, and if so, it initializes the authentication client control. If the authentication client control is not found, the log-on proceeds as normal, e.g., the client returns a user name and password, and logs in. If, however, the authentication client control is found, the process below is followed.

The client 240 initiates a session with the authentication server. (message 3). For one embodiment, the session is initiated via HTTPS, or another secure mechanism. For one embodiment, this process is driven by a client authentication object.

The authentication server sends a nonce to client object (message 4). The nonce, for one embodiment, is a large random number. For one embodiment, the nonce may include certain identification data within the number, such as a time/date stamp or similar data. For one embodiment, the nonce may further include data regarding the IP address of the client 240.

For one embodiment, the client authentication object raises event to indicate that it is ready. The logon page alerts the user that the fingerprint sensor is ready. The user performs the biometric authentication. For one embodiment, the user places the finger on the sensor, to use a fingerprint.

The client-side software obtains the biometric data, and performs feature extraction to generate a template. The client-side software then combines this

template with the nonce that came from the authentication server. (block 5). For another embodiment, the client 240 does not perform the feature extraction, but rather combines the nonce with the image obtained.

The client authentication object obtains a client username. For one embodiment, this data is obtained from the HTML page. For one embodiment, this is obtained by the client authentication object raising an event. For one embodiment, if the page does not return this information, the client object may request this information from the user. This data is then sent to the partner server 230. (message 6). For one embodiment, this is sent via an HTTP POST. Note that this data is not sent to the authentication server 220.

In response to receiving this data, the web service 230 generates a challenge, e.g. one-time password, and encrypts it using the public key associated with the username. The web server 230 further obtains the record ID associated with the username (block 7). For one embodiment, this process is driven by the application web server. For one embodiment, the encryption is performed by JavaBean supplied by the authentication system, when the partner 230 registers to accept third party authentication.

The web service sends the record ID associated with the username and the encrypted challenge to the client 230. (message 8). For one embodiment, the web server also sends the policy to the client 230. For one embodiment, if the policy requires some data directly from the client 230, such as a password, this data must first be supplied, prior to receiving the challenge. Thus, there would be an additional exchange, passing the requested password/other authentication data to the partner 230.

After that level of validation occurs, the partner 230 passes the encrypted challenge to the client 230. For another embodiment, the policy may involve additional verification by the third party authentication server 220, such as additional biometric data or a password maintained by the authentication server 220. In that instance, the policy data would be included with the challenge. For one embodiment, the policy is encrypted with the challenge, so the client 240 could not access the policy. This process, for one embodiment, is driven by application web server.

The client object forwards the record ID, encrypted challenge, and if appropriate the policy, to the authentication server. The client object also sends the encrypted biometric template to the authentication server. (message 9).

The authentication server 220 compares the biometric template received from the client 240 against the template associated with the record ID. The authentication server 220 then determines if the policy requires additional data. For example, the policy may require multiple biometric matches to authenticate. The authentication server 220 follows the policy defined by the web server 230, and only declares a match if all the data necessary for a match has been obtained. If a valid match is found, the authentication server 220 decrypts the challenge with the private key 525 associated with that record ID. (block 10).

The authentication server sends the decrypted challenge to the client object. (message 11). For one embodiment, as discussed above, this occurs over a secure channel. For one embodiment, the decrypted challenge is encrypted with the partner key.

The client object passes the challenge on to the web service (message 12). The web service compares challenge received to the challenge sent (block 13). If

the challenges match, and all other aspects of the policy have been satisfied, the web service permits the user to access the partner. At this point, the user has been successfully validated.

Figure 6A is a flowchart of one embodiment enabling a client to use the authentication server. The process starts at block 605, when the client creates a connection to the authentication server, in order to create an account. The software to administer the client registration is downloaded to the client.

At block 610, a secure session is created between the client and the server.

At block 615, the authentication server creates a nonce to send to the client. The nonce is used to prevent replay attacks.

At block 620, the client, on instruction from the authentication server, captures biometric data. For one embodiment, multiple sets of biometric data may be captured. For example, if the biometric data is a fingerprint, then multiple fingers may be registered at this point.

The software downloaded to the client may extract the features of the biometric data, and create a biometric template. For a fingerprint, this may be a list of minutiae.

At block 625, the biometric template is encrypted with/combined with the nonce, and returned to the authentication server. For another embodiment, if the client does not perform the biometric feature extraction, the captured biometric is combined with the nonce, and returned at this point.

At block 630, the authentication server extracts the nonce, and verifies it. If the nonce is successfully verified, the biometric template is obtained at this point.

At block 635, an anonymous record is created for the user. The anonymous record includes the user's biometric data.

At block 640, a record ID is generated for the anonymous record. For one embodiment, the record ID is generated randomly. For another embodiment, record IDs may be sequential, or may be generated using some other mechanism.

A public/private key pair is also generated for the client. For one embodiment, the public/private key pair may be a maximum length. For another embodiment, multiple key pairs may be generated, depending on export restrictions. For one embodiment, the public key is certified by a certification authority. The process of certifying a public key is known in the art. For one embodiment, the certification authority may be within the authentication server itself. For another embodiment, an external authentication server may be contacted at this point to certify the public key.

At block 645, an entry is created in the credential database. The entry is indexed by the record ID, and includes the biometric template(s) and the private key(s) of the user.

At block 650, the record ID and public key(s) are returned to the client. If the public key(s) have been certified, the certified key(s) are returned to the client.

At block 655, the client stores the public key(s) and record ID. For one embodiment, the client only stores this data temporarily, until it is passed on to the partner site, as will be described below.

Figure 6B illustrates the process of adding the credential data to a partner site. The process starts when the client either first logs into the partner site, or

first logs into the partner site after receiving the credential data from the authentication server.

At block 660, the client connects to the partner site. If the client already has an account with the partner site, the standard log-in is performed. If this is an initial registration, the partner site at this point collects all relevant information. This corresponds to the "existing entries" portion of the database.

The client also indicates that it has credential data with the third party authentication server.

At block 665, the process determines whether the partner site is enabled to handle such third-party authentication. If the partner site is enabled, the process continues to block 675. Otherwise, at block 670, the log-on process is completed, and the user can continue to use the partner site, as normal.

At block 675, the client passes the record ID and the public key to the partner.

At block 680, the partner creates the additional, third party authorization specific entries in the database. These entries include the client ID (record ID), and client public key.

At block 685, the partner associates a policy with the client entry. The policy determines what authentication(s) take place to permit a connection between the client and the server. The policy may determine the combination of username, biometrics, passwords, and other items such as smart cards that should be used to authenticate the user.

The process then continues to block 670, and the log-on process is completed, and the user can continue to use the partner site. For one

embodiment, the login process described above with respect to Figure 5 is performed here.

Figure 7 is a flowchart of one embodiment of setting up a partner site to use the authentication server. The process starts at block 710, when the partner 230 connects to the authentication server 220, or the server that permits the partner 230 to download the appropriate data.

At block 720, the software and/or data for this operation is downloaded to the partner.

At block 730, the registration and login code is modified, to permit the use of third-party authentication. For one embodiment, this alteration is to HTML code. For one embodiment, this alteration includes a script, which detects the presence of a sensor, and permits the user to use the sensor. The script further sends the challenge, and receives the challenge.

At block 740, fields are added to the user database. For one embodiment, the fields include a record ID and a public key, received from the user. Furthermore, the fields may include a policy, which indicates what authentication level is needed for access. For example, the policy may require multiple biometrics, or a biometric and a password to access the partner. This policy, or a pointer to the appropriate policy, is added to the client data field.

For one embodiment, the fields further include an area to save the challenge, which is described above. This permits the easy association of the challenge with the particular user. Thus, when the user returns the record ID (client ID), with the decrypted challenge, the partner knows which client this refers to, and the appropriate challenge.

At block 750, the process adds logic to generate the challenge, and to compare the data returned with the challenge, to determine access.

At block 760, the partner key is exchanged with the authentication server. For one embodiment, the partner key is a symmetric key, or a plurality of symmetric keys. For another embodiment, the partner key is a pair of public/private key sets, one each of the partner and the authentication server, to permit secure communications between the partner and the authentication server.

The process then ends, at block 770. At this point, if a client is enabled to use third party authentication, the partner is able to use that ability, as was described above.

In the foregoing specification, the invention has been described with reference to specific exemplary embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention as set forth in the appended claims. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense.

CLAIMS

What is claimed is:

1. A method of authenticating a client, the method comprising:

receiving a record ID for a user, and a one-time key generated by the server and encrypted with a user's public key by the server;

receiving the user's authentication data from the client;

determining if the user's authentication data matches the record ID; and

if so, decrypting the one-time key with the user's private key, and returning the decrypted one-time key to the client.

2. The method of claim 1, further comprising registering the user, registering comprising:

receiving a registration authentication data from the user;

generating a random public key/private key pair for the user;

generating a random record ID for the user; and

associating the authentication data and the private key with the record ID.

3. The method of claim 2, further comprising:

sending the record ID and the public key to the user.

4. The method of claim 2, further comprising establishing a secure connection with the user, prior to receiving registration authentication data.

5. The method of claim 1, wherein a web page presented by the server to the client prompts the user to enter the authentication data to log in to the server.

6. The method of claim 5, wherein the client's authentication data is automatically redirected to the authentication server.

7. The method of claim 1, wherein the authentication data is biometric data.

8. The method of claim 1, wherein the authentication data is personal data selected from among the following: a password, a smart card, and another type of authentication card.

9. The method of claim 1, wherein the client forwards the decrypted one-time key to the server, thereby authenticating the user as the owner of the private key.

10. The method of claim 1, further comprising discarding the record ID after returning the one-time key to the user.

11. The method of claim 1, wherein the record ID and the encrypted one-time key are further encrypted using a partner key, the method further comprising decrypting the record ID and encrypted one-time key using the partner key.

12. The method of claim 11, wherein the partner key is a symmetric key set up during registration of the partner.

13. The method of claim 11, wherein the partner key is a private key of the authentication server.

14. A method of using a third party authentication server to authenticate a user to a server, the method comprising:

looking up a record ID associated with the user;

generating a one-time key and encrypting the one-time key with a public key of the user, and sending the encrypted one-time key and the record ID to the user;

receiving authentication data, the authentication data being the decrypted one-time key; and

permitting access to the server.

15. The method of claim 14, comprising:

determining an authentication policy associated with the user; and

verifying that the authentication policy has been satisfied, prior to permitting access to the server.

16. The method of claim 15, wherein verifying that the authentication policy has been satisfied comprises:

determining if the server should verify additional data; and

if so, requesting additional data from the user prior to generating the one-time key.

17. A third-party authentication system comprising:

an authentication server to receive a record ID for a user, and a one-time key generated by the server and encrypted with a user's public key by the server;

a comparison logic to receive user authentication data from the client and comparing whether the user's authentication data matches the record ID; and

a decryption logic to decrypt the one-time key with a private key associated with the validated record ID, and returning the decrypted one-time key to the client.

18. The system of claim 17, further comprising:

a policy validation logic to receive a policy from the server, and determine if the policy has been fulfilled; and

the decryption logic to decrypt the one-time key only if the policy has been fulfilled.

19. The system of claim 17, further comprising:

a nonce generation logic to generate a nonce, the nonce to be included with the user authentication data from the client; and

the comparison logic to verify that the user authentication data includes the appropriate nonce.

20. The system of claim 17, further comprising a client registration logic to register the user, the client registration logic comprising:

a key generation logic to generate a random public key/private key pair for the user;

a record ID generation logic to generate a random record ID for the user; and

the client registration logic to associate user authentication data with the private key and the record ID.

21. The system of claim 18, further comprising:

the interface to send the record ID and the public key to the user.

22. The system of claim 19, wherein the interface establish a secure connection with the user, prior to receiving registration authentication data.

23. The system of claim 17, wherein a web page presented by the server to the client prompts the user to enter the authentication data to log in to the server.

24. The system of claim 23, wherein the client's authentication data is automatically redirected to the authentication server.

25. The system of claim 17, wherein the authentication data is biometric data.

26. The system of claim 17, wherein the authentication data is personal data selected from among the following: a password, a smart card, and another type of authentication card.

27. The system of claim 17, wherein the client forwards the decrypted one-time key to the server, thereby authenticating the user as the owner of the private key.

28. The system of claim 17, further comprising a security mechanism to discard the record ID after returning the one-time key to the user.

29. The system of claim 17, wherein the decryption logic further decrypts the record ID and the encrypted one-time key with a partner key.

30. The system of claim 29, wherein the partner key is a symmetric key set up during registration of the partner.

31. The system of claim 29, wherein the partner key is a private key of the authentication server.

ABSTRACT OF THE DISCLOSURE

A method and apparatus for a third party authentication server is described. The method includes receiving a record ID for a user, and a one-time key generated by the server and encrypted with a user's public key by the server. The method further includes receiving the user's authentication data from the client, and determining if the user's authentication data matches the record ID. If the authentication data matches the record ID, decrypting the one-time key with the user's private key, and returning the decrypted one-time key to the client.

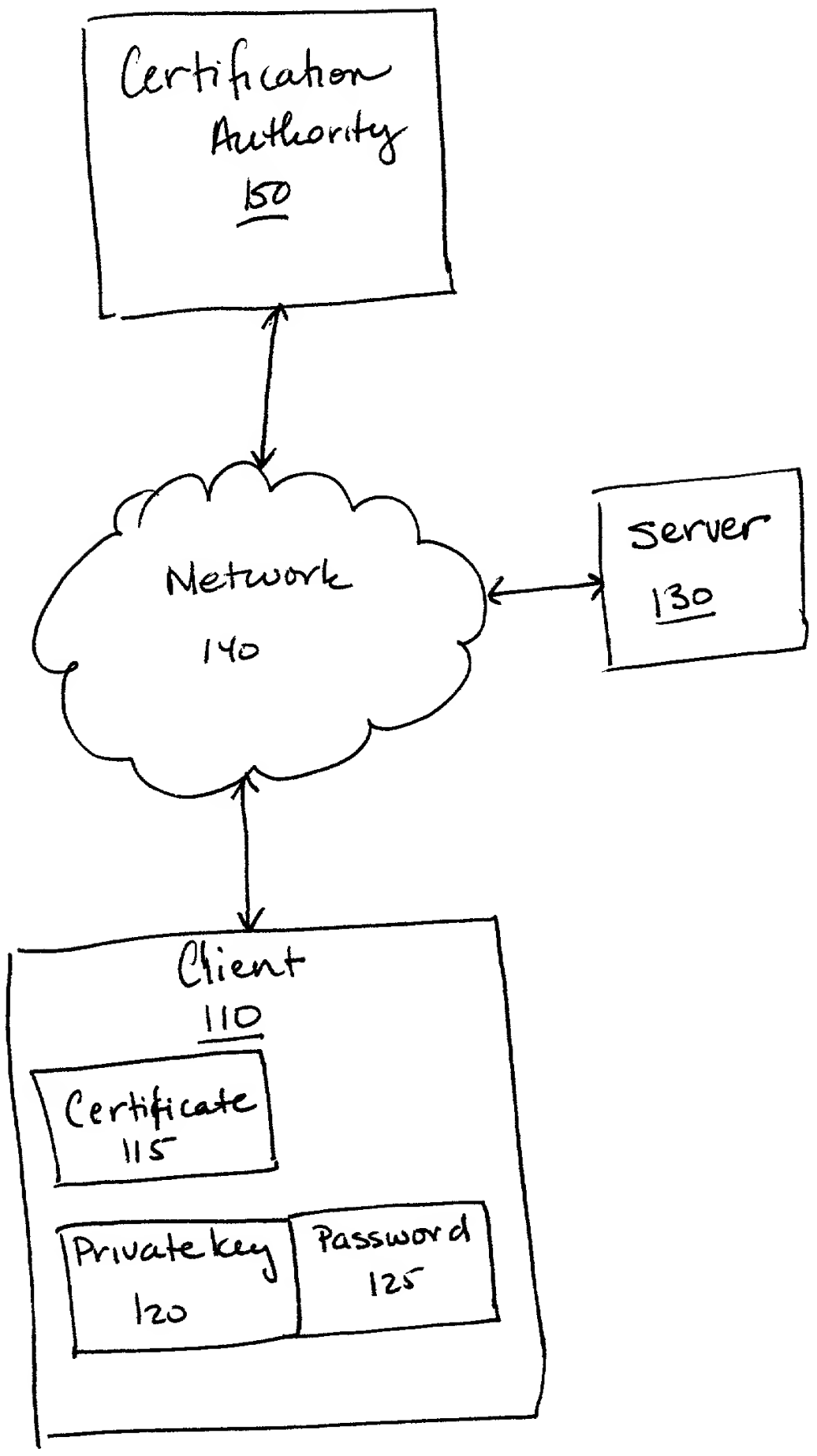

FIG. 1 (PRIOR ART)

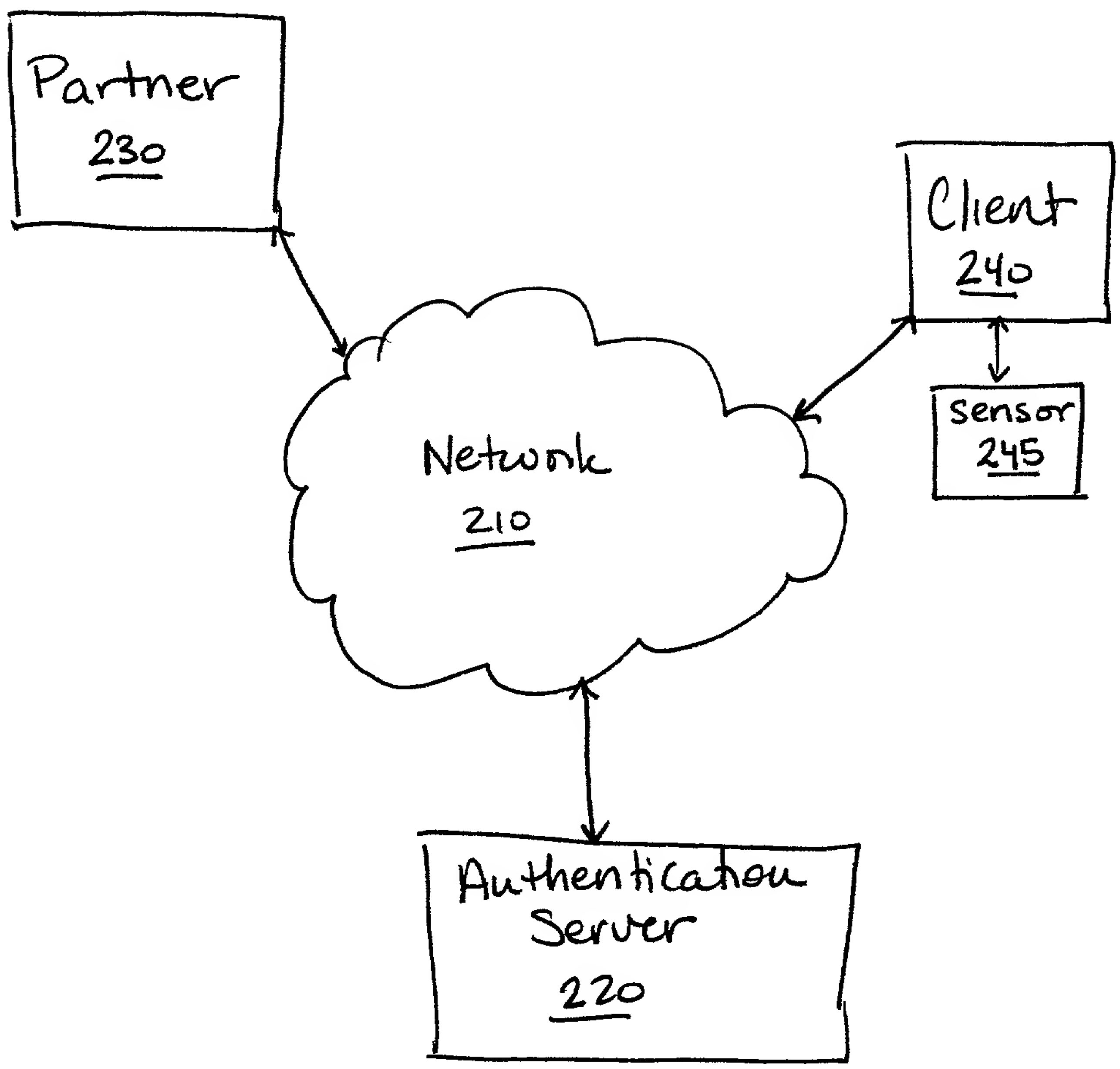
Partner
230
Client
240
Sensor
245
Network
210
Authentication Server
220

FIG. 2

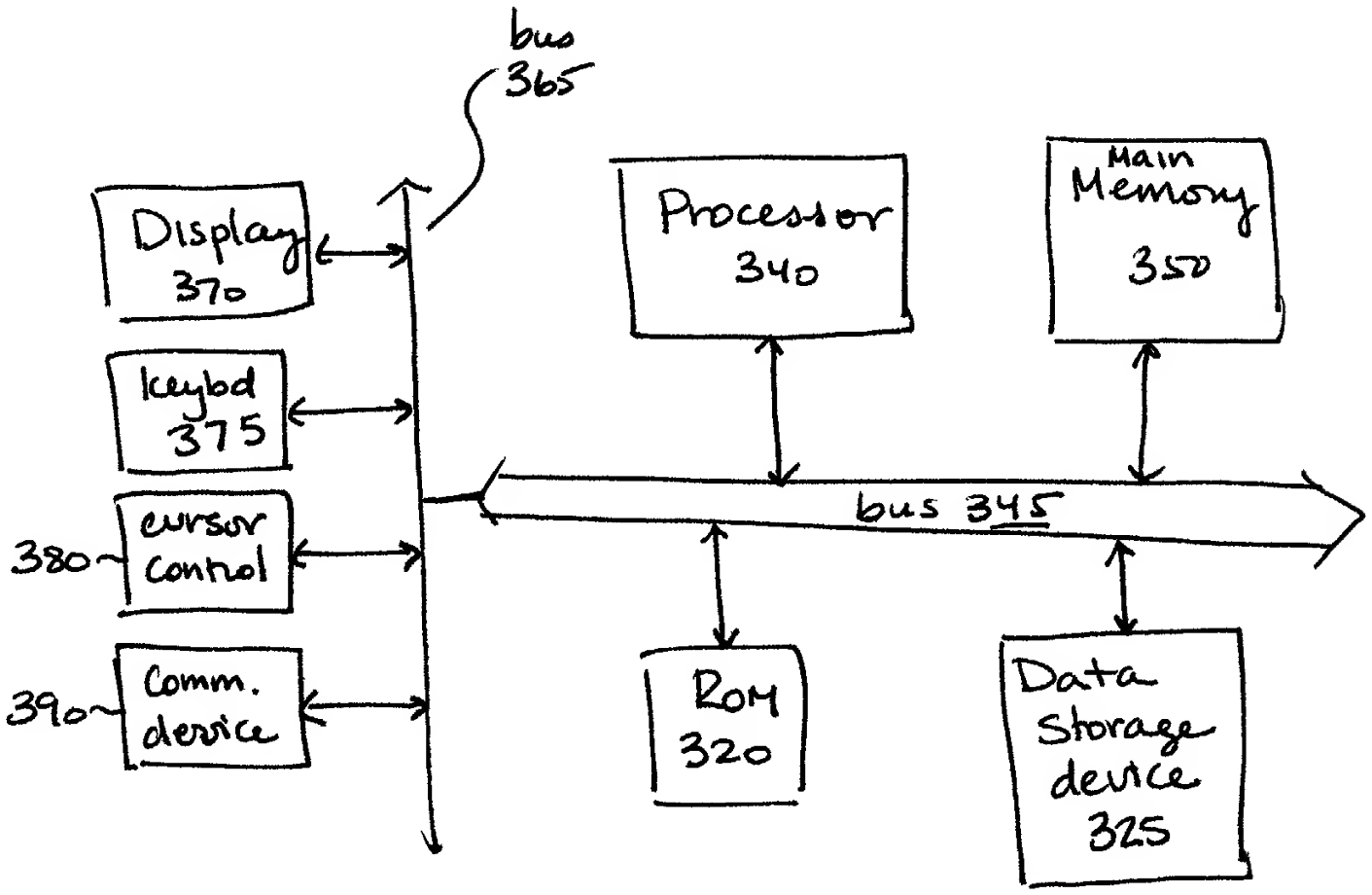
bus
365
Display
370
Keybd
375
380
cursor
control
390
Comm.
device
Processor
340
Main
Memory
350
bus 345
ROM
320
Data
Storage
device
325

FIG. 3

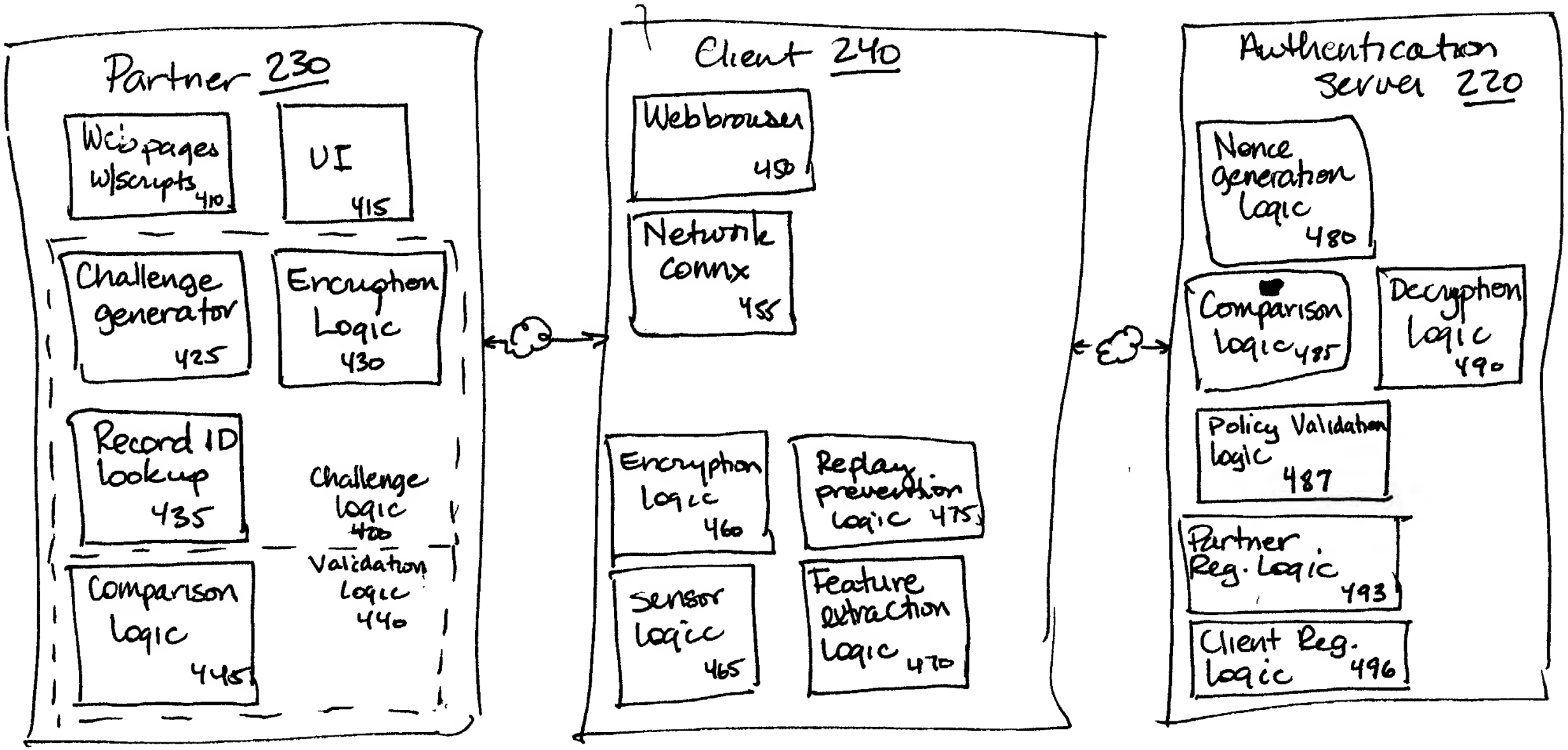
Partner 230
Web pages w/scripts 410
UI 415
Challenge generator 425
Encryption Logic 430
Record ID lookup 435
Challenge logic 420
Validation logic 440
Comparison logic 445
Client 240
Webbrowser 450
Network comm 455
Encryption Logic 460
Replay prevention Logic 475
Sensor Logic 465
Feature extraction Logic 470
Authentication Server 220
Nonce generation Logic 480
Comparison Logic 485
Decryption Logic 490
Policy Validation logic 487
Partner Reg. Logic 493
Client Reg. Logic 496

Fig 4

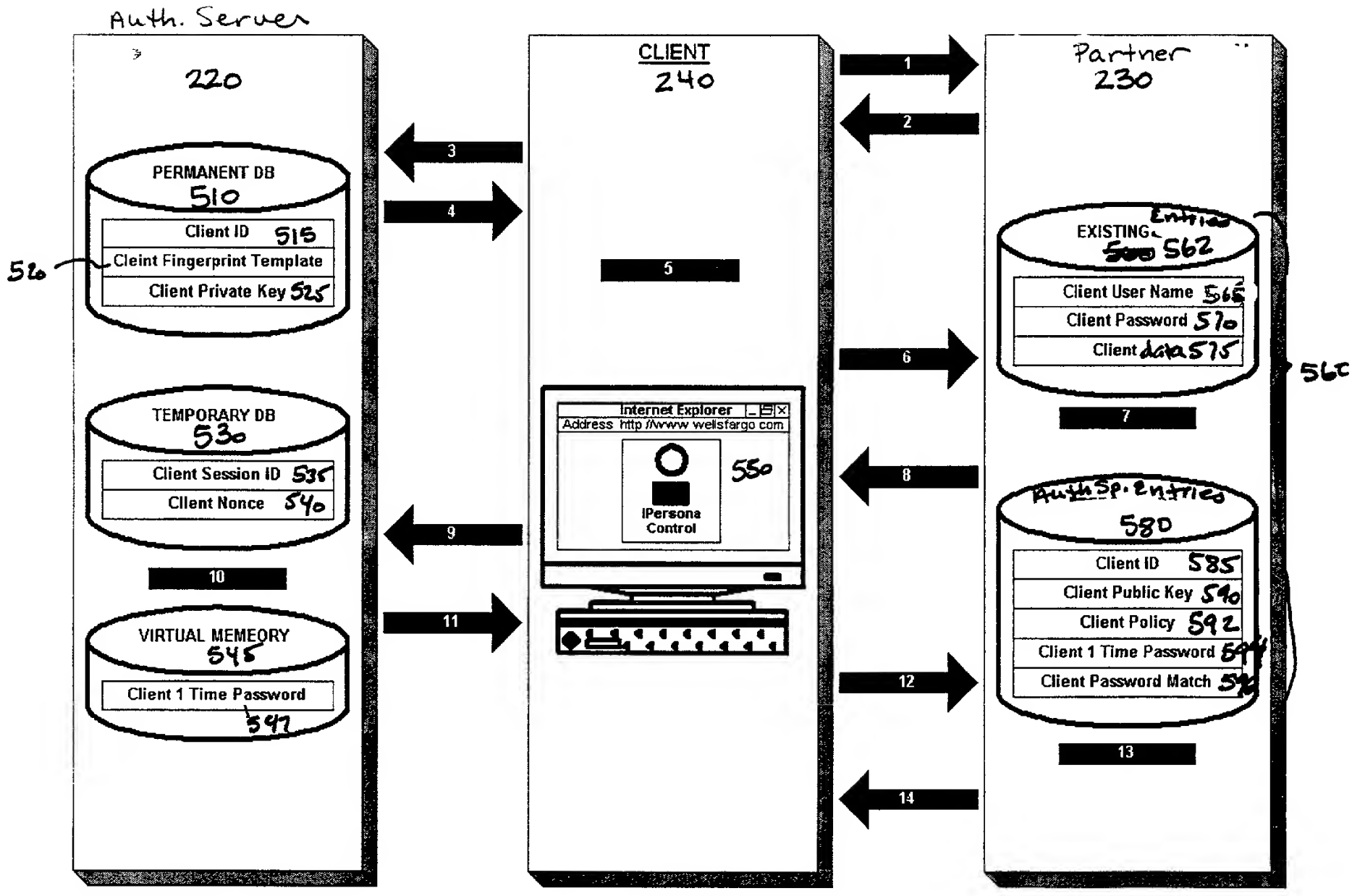
Auth. Server
220
PERMANENT DB
510
Client ID 515
Cleint Fingerprint Template
Client Private Key 525
526
TEMPORARY DB
530
Client Session ID 535
Client Nonce 540
10
VIRTUAL MEMEORY
545
Client 1 Time Password
547
CLIENT
240
5
Internet Explorer
Address http://www.wellsfargo.com
IPersona Control
550
Partner
230
EXISTING Entities
500 562
Client User Name 565
Client Password 570
Client data 575
560
7
AuthSp. Entities
580
Client ID 585
Client Public Key 590
Client Policy 592
Client 1 Time Password 594
Client Password Match 596
13
1
2
3
4
6
8
9
11
12
14

FIG. 5

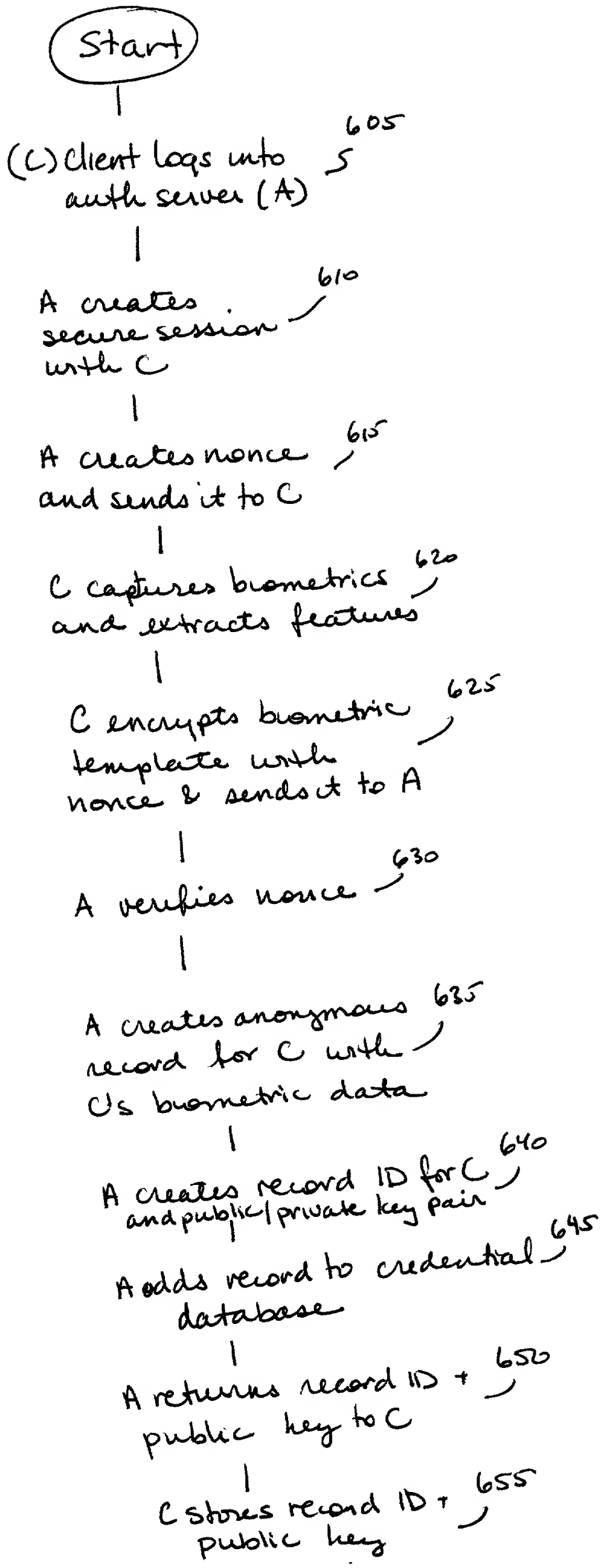
Start
(C) client logs into auth server (A)
605
A creates secure session with C
610
A creates nonce and sends it to C
615
C captures biometrics and extracts features
620
C encrypts biometric template with nonce & sends it to A
625
A verifies nonce
630
A creates anonymous record for C with Cs biometric data
635
A creates record ID for C and public/private key pair
640
A adds record to credential database
645
A returns record ID + public key to C
650
C stores record ID + public key
655

FIG. 6A

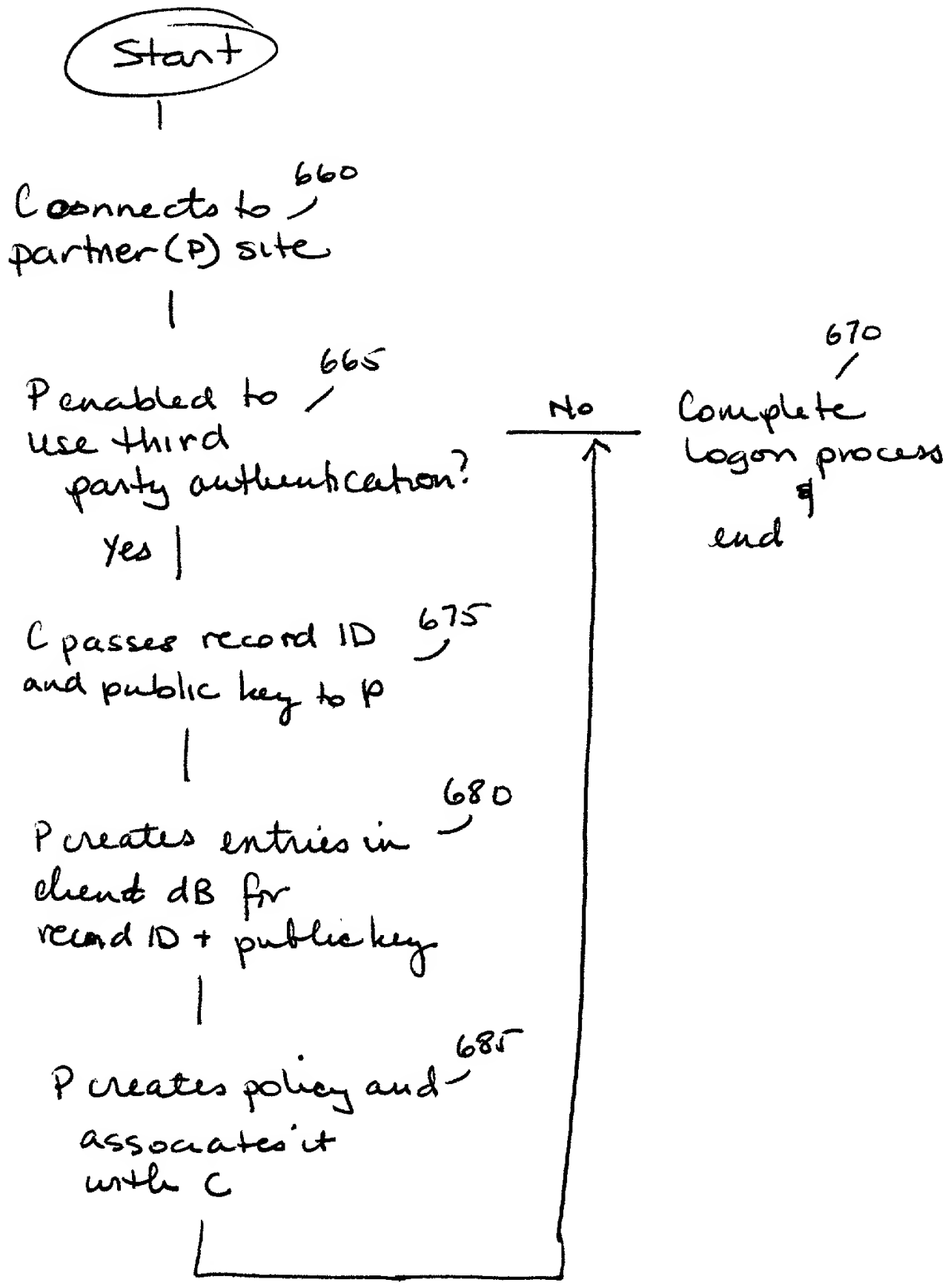
Start
C connects to partner (P) site
660
P enabled to use third party authentication?
665
No
Complete logon process
end
670
Yes
C passes record ID and public key to P
675
P creates entries in client dB for record ID + public key
680
P creates policy and associates it with C
685

FIG. 6B

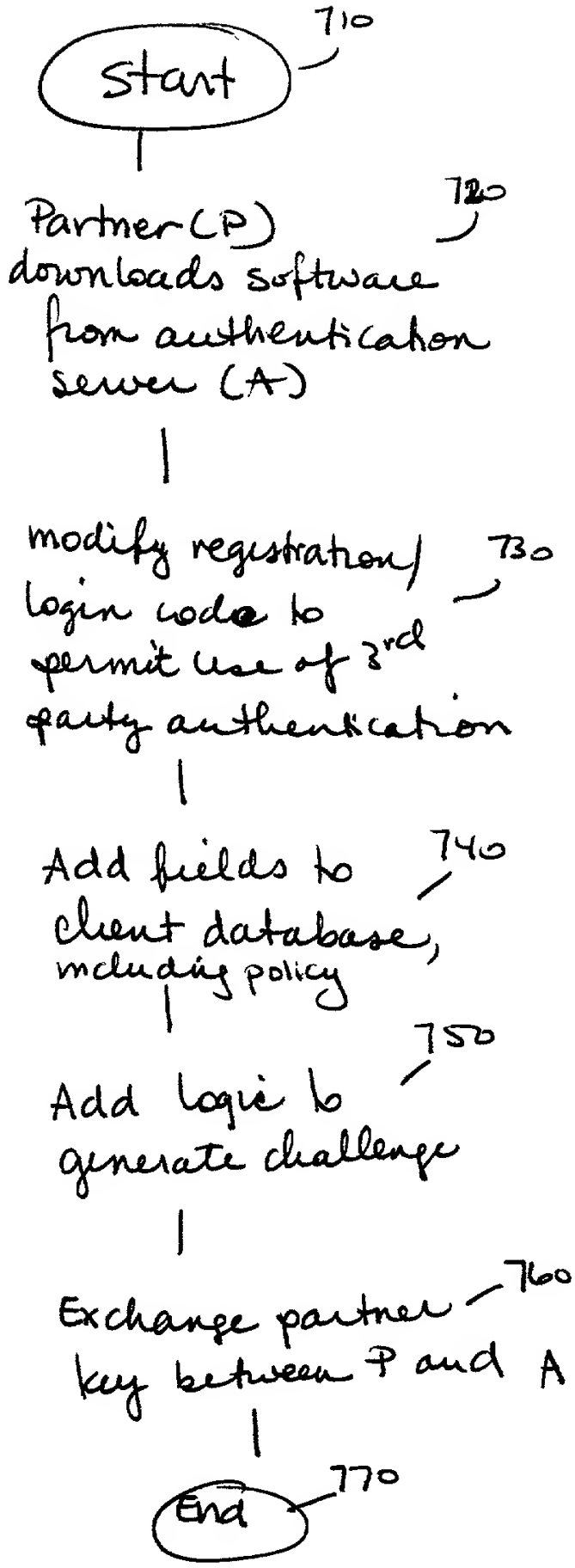
Start
710
Partner (P) downloads software from authentication server (A)
720
modify registration/ login code to permit use of 3rd party authentication
730
Add fields to client database, including policy
740
Add logic to generate challenge
750
Exchange partner key between P and A
760
End
770

FIG. 7